High Dimensional Error-Correcting Codes

Eitan Yaakobi
Dept. of Electrical and Computer Engineering
University of California, San Diego
La Jolla, CA 92093, USA
Email: eyaakobi@ucsd.edu

Tuvi Etzion
Dept. of Computer Science
Technion-Israel Institute of Technology
Haifa 32000, Israel
Email: etzion@cs.technion.ac.il

***Abstract*—In this paper we construct multidimensional codes with high dimension. The codes can correct high dimensional errors which have the form of either small clusters, or confined to an area with a small radius. We also consider small number of errors in a small area. The clusters which are discussed are mainly spheres such as semi-crosses and crosses. Also considered are clusters with small number of errors such as 2-bursts, two errors in various clusters, and three errors on a line. Our main focus is on the redundancy of the codes when the most dominant parameter is the dimension of the code.**

I. Introduction

Multidimensional coding in general and two-dimensional coding in particular is a subject which attracts a lot of attention in the last three decades. But, although the related theory of the one-dimensional case is well developed, the theory for the multidimensional case is developed rather slowly. This is due to the fact that most of the one-dimensional techniques are not generalized easily to higher dimensions and usually many different techniques are used in the multidimensional case.

Remark 1: In our discussion we will consider noncyclic arrays, even if the construction works on a cyclic array, i.e., a torus. This is done for convenient reasons. In the following redundancy definition, the array is considered to be cyclic. But, since the size of the array is very large we will omit the minor difference in the redundancy between a cyclic array and a noncyclic array.

A binary multidimensional error-correcting code corrects errors which occur in a multidimensional array. Throughout the paper the volume of the array is N. If we are given a set with β possible patterns of errors (no error is also such a pattern) that can occur anywhere in the array then the redundancy of the code must satisfy $r \geq \log(N \cdot \beta) = \log N + \log \beta$ (all logarithms in this paper are in base 2). The difference $r - \log N$ is called the *excess redundancy* of the code [1], [2].

Abdel-Ghaffar [1] constructed binary two-dimensional codes which correct a cluster of a rectangle shape with area B for which $r = \lceil \log N \rceil + B$. These codes attain the lower bound on the excess redundancy. There is no known generalization for the construction of Abdel-Ghaffar [1] to more than two dimensions. Moreover, the number of length parameters on which the construction works is very limited. A construction in [3] produces a D-dimensional code for correction of a D-dimensional box-error with redundancy $\lceil \log N \rceil + B + \lceil \log b_1 \rceil$, where b_1 is the length of the D-dimensional box in the first dimension. For the two-dimensional case, this construction is more flexible in its parameters than the construction in [1].

In this paper we are interested in D-dimensional codes, where $D > 2$ is usually very large. On the other hand, we are interested either in a small number of errors or that the cluster is spread in radius at most one from the center of the error event.

How can we correct such bursts? If the size of the burst is one then we can always use an one-dimensional Hamming code folded on the D-dimensional array to obtain an optimal code. Given a set S with patterns of errors, the most natural and simple way to correct an error from S is to correct a box-error which contains all possible errors from S. This might result in a large excess redundancy as the number of patterns in S might be much smaller than the number of patterns defined by a box-error. The goal of this paper is to construct codes with excess redundancy much smaller than the one implied by a correction of a related box-error. If the size of the burst is two then we will see in the sequel that a code with optimal, or almost optimal, excess redundancy can be constructed. But, if the size of the burst is three then we don't know how to construct a code which attains the lower bound on the excess redundancy. In fact, the question how to construct an optimal code which corrects an arbitrary cluster is still open.

The rest of the paper is organized as follows. In Section II we present the definitions of linear codes and shapes which are discussed throughout this paper. In Section III we discuss codes for which the error is small and confined to an one-dimensional line. We will examine two types of errors, 2-bursts and 3-bursts (bursts of length two and three, respectively) on a line, where a *b-burst* is any set of errors that is confined to an area of size b. In Section IV we discuss a coloring method presented in [3] and explain how it is designed to correct cluster errors. We show how to correct error whose shape is a semi-cross (corner) with arms of length one (radius of length one) or a cross (Stein's sphere) with arms of length one (radius of length one). These two shapes will exhibit an excellent example for the strength of the coloring method. In Section V we consider clusters with small weight inside a relatively larger cluster. We present asymptotically optimal solutions for the case where the weight is two and the cluster is a semi-cross, cross, or a two-dimensional square. In Section VI we conclude and present the goals for the future research.

II. Basic Definitions

A binary multidimensional b-error-correcting code is a set C consisting of D-dimensional binary arrays of the same size, such that if we are given an array $\mathcal{A}$ from C and the values of up to b positions in $\mathcal{A}$ are changed, then we will be able to recover $\mathcal{A}$. We consider only linear codes as done in all previous works.

A binary D-dimensional linear code C is a linear subspace of the $n_1 \times n_2 \times \cdots \times n_D$ binary matrices. If the subspace is of dimension $N - r$, where $N = \prod_{\ell=1}^{D} n_\ell$, we say that the code is an $[n_1 \times n_2 \times \cdots \times n_D, N - r]$ code. The code can be also defined by its parity-check matrix. Let $H = (h_{\mathbf{i},j})$, where $\mathbf{i} \in \mathbf{I}$, $\mathbf{I} = \{(i_1, i_2, \ldots, i_D) \ : \ 0 \le i_\ell \le n_\ell - 1\}$, and $0 \le j \le r-1$, be a $(D+1)$-dimensional binary matrix of size $n_1 \times n_2 \times \cdots \times n_D \times r$, consisting of r linearly independent $n_1 \times n_2 \times \cdots \times n_D$ matrices. Let $c = (c_\mathbf{i})$ denote a binary $n_1 \times n_2 \times \cdots \times n_D$ matrix. The linear subspace defined by the following set of r equations,

$$\sum_{\mathbf{i} \in \mathbf{I}} c_\mathbf{i} h_{\mathbf{i},j} = 0,$$

for all $0 \le j \le r-1$, is an $[n_1 \times n_2 \times \cdots \times n_D, N - r]$ code. We say that r is the redundancy of the code.

Our goal in this paper is to handle D-dimensional errors from one of the following types:

- Errors which don't spread more than one position around an artificial center (which is the center of the error event).
- Two errors in a cluster of some shape.

These clusters include the following types of bursts.

1) A D-dimensional 2-burst which corresponds to any two adjacent positions that might be in error.
2) A D-dimensional 3-burst in which all the errors are on the same line. Such an error corresponds to three positions of the form $(i_1, \ldots, i_{j-1}, i_j - 1, i_{j+1}, \ldots, i_D)$, $(i_1, \ldots, i_{j-1}, i_j, i_{j+1}, \ldots, i_D)$, and $(i_1, \ldots, i_{j-1}, i_j + 1, i_{j+1}, \ldots, i_D)$ for some j, $1 \le j \le D$.
3) A D-dimensional burst whose shape is a semi-cross with arms of length one. Such a semi-cross has a center point at $(i_1, i_2, \ldots, i_D)$ and includes all the points of the form $(i_1, \ldots, i_{j-1}, i_j + 1, i_{j+1}, \ldots, i_D)$, $1 \le j \le D$.
4) A D-dimensional burst whose shape is a cross with arms of length one. Such a cross has a center point at $(i_1, i_2, \ldots, i_D)$ and includes all the points of the form $(i_1, \ldots, i_{j-1}, i_j - 1, i_{j+1}, \ldots, i_D)$ and all the points of the form $(i_1, \ldots, i_{j-1}, i_j + 1, i_{j+1}, \ldots, i_D)$, $1 \le j \le D$.
5) Two errors inside a semi-cross or a cross with arms of length R. A semi-cross with arms of length R has a center point at $(i_1, i_2, \ldots, i_D)$ and includes all the points of the form $(i_1, \ldots, i_{j-1}, i_j + \ell, i_{j+1}, \ldots, i_D)$, $1 \le j \le D$, $1 \le \ell \le R$. Similarly, a cross with arms of length R is defined. These errors are also related to two errors inside a two-dimensional square with edges of length R.

Why are we interested in crosses and semi-crosses? Errors are likely to be spread within spheres to some limited radius. Crosses and semi-crosses are types of spheres as described in [4] which are relatively simpler to handle than other spheres. These spheres are also discussed extensively in the literature, e.g. [5], [6], [7], [8].

III. Constructions with Low Redundancy

In this section we will handle two types of errors, 2-burst and 3-burst on a straight line. The number of possible patterns of errors (excluding no errors) which can be confined to a D-dimensional 2-burst is $D + 1$ and to a D-dimensional 3-burst on a line is $3D+1$. Hence, a lower bound of their redundancies is log $((D + 2) \cdot N)$ and log $((3D + 2) \cdot N)$, respectively.

A. Correction of 2-burst

Assume that we have a D-dimensional array of size $n_1 \times n_2 \times \cdots \times n_D$ on which we want to correct any cluster of error that can be confined to a 2-burst.

Construction A: Let α be a primitive element in GF(2^m) for $2^m - 1 \ge \prod_{\ell=1}^{D} n_\ell$. Let $d = \lceil \log D \rceil$ and $\mathbf{i} = (i_1, i_2, \ldots, i_D)$, where $0 \le i_\ell \le n_\ell - 1$. Let A be a $d \times D$ matrix containing distinct binary d-tuples as columns. We construct the following $n_1 \times n_2 \times \cdots \times n_D \times (m + d + 1)$ parity check matrix H:

$$h_\mathbf{i} = \begin{bmatrix} 1 \\ A\mathbf{i}^T \mod 2 \\ \alpha^{\sum_{j=1}^{D} i_j (\prod_{\ell=j+1}^{D} n_\ell)} \end{bmatrix},$$

for all $\mathbf{i} = (i_1, i_2, \ldots, i_D)$, where $0 \le i_\ell \le n_\ell - 1$.

Remark 2: The matrix A that we are using was also used in [9], but the construction here is more flexible in its parameters. This is a consequence from the way that we fold the elements of GF(2^m) into the parity-check matrix of the code. Moreover, we will see in the sequel that this method will be of use for constructions of larger bursts with at most two erroneous positions.

In the decoding algorithm we assume that the error occurred can be confined to a 2-burst. The syndrome received v in the decoding algorithm consists of three parts.

- The first bit determines the number of errors occurred. Obviously if the syndrome is the all-zeroes vector than no errors occurred. If the first bit of the syndrome is an *one* then exactly one error occurred and its position is the position of v in H. If the first bit of a non-zero vector v is a *zero* then two errors occurred. Their position is determined by the other $m + d$ entries of v.
- The next d bits determine the dimension in which the burst occurred. There are D dimensions and each column of the matrix A corresponds to a different dimension for two consecutive errors. If the errors occurred in positions $\mathbf{i}_1 = (i_1, \ldots, i_D)$ and $\mathbf{i}_2 = (i_1, \ldots, i_{j-1}, i_j + 1, i_{j+1}, \ldots, i_D)$ then the value of the d bits, $(A\mathbf{i}_1^T + A\mathbf{i}_2^T) \mod 2$, is the j-th column of the matrix A.
- The entries of the last m rows of the matrix H form the folding of the first $\Pi_{\ell=1}^{D} n_\ell$ consecutive elements of GF(2^m). Given a dimension ℓ there exists an integer $i(\ell)$ such that each two consecutive elements in dimension ℓ have the form $\alpha^j, \alpha^{j+i(\ell)}$. It is easy to verify that for $j_1 \ne j_2$ we have $\alpha^{j_1} + \alpha^{j_1+i(\ell)} \ne \alpha^{j_2} + \alpha^{j_2+i(\ell)}$. Thus, given the dimension of the burst of size two, the last m bits of v can determine the two consecutive positions of the burst.

Theorem 1:

- The code constructed in Construction A can correct any error pattern confined to a 2-burst in an $n_1 \times n_2 \times \cdots \times n_D$ array codeword.
- The code constructed by Construction A has redundancy which is greater by at most two from the trivial lower bound on the redundancy.

Remark 3: There are cases in which we can prove that the code of Construction A is optimal.

B. 3-burst on a line

Next, we would like to consider correction of error patterns confined to an arbitrary D-dimensional 3-burst. This appears to be much more difficult than the 2-burst case. The main reason is that the error can be spread on a two-dimensional subspace. Therefore, we consider only the case of a D-dimensional cluster of size three on an one-dimensional subspace, i.e. on a straight line. In this case we can generalize Construction A.

Construction B: Let α be a primitive element in GF(2^m) for $2^m-1 \geq \prod_{\ell=1}^{D} n_\ell$. Let B be a matrix of size $\lceil \log (D+1) \rceil \times D$ which contains all binary representations of the integers between 1 and D as its columns, such that the binary representation of D is the left most column, and the binary representation of 1 is the right most column. The rows of B are denoted by $b_1, b_2, \ldots, b_{\lceil \log (D+1) \rceil}$. Let β be a primitive element in GF(4). By abuse of notation, if $\mathbf{v} = (v_1, v_2, \ldots, v_{\lceil \log (D+1) \rceil})^T$ is a column vector of length $\lceil \log (D+1) \rceil$, we denote $\beta^{\mathbf{v}} = (\beta^{v_1}, \beta^{v_2}, \ldots, \beta^{v_{\lceil \log (D+1) \rceil}})^T$. We construct the following $n_1 \times n_2 \times \cdots \times n_D \times (m + 2\lceil \log (D+1) \rceil + 2)$ parity check matrix H^D:

$$h_{\mathbf{i}}^D = \begin{bmatrix} 1 \\ \left(\sum_{j=1}^{D} i_j\right) \bmod 2 \\ \beta^{B\mathbf{i}^T} \\ \alpha^{\sum_{j=1}^{D} i_j (\prod_{\ell=j+1}^{D} n_\ell)} \end{bmatrix},$$

for all $\mathbf{i} = (i_1, i_2, \ldots, i_D)$, where $0 \leq i_\ell \leq n_\ell - 1$. The multiplication $B\mathbf{i}^T$ is taken over the integers and the vector $\beta^{B\mathbf{i}^T}$ consists of $\lceil \log (D+1) \rceil$ vectors of length two, each one representing an element in GF(4).

Theorem 2:

- The code constructed in Construction B can correct any error pattern confined to a 3-burst on a straight line in an $n_1 \times n_2 \times \cdots \times n_D$ array codeword.
- The code constructed by Construction B has excess redundancy $2\lceil \log (D+1) \rceil + 2$ which is at most twice than the trivial lower bound on the excess redundancy.

IV. The Coloring Method

The coloring method introduced in [3] is an effective method to handle multidimensional cluster errors. In the coloring method we use D one-dimensional auxiliary codes for our encoding and decoding procedures. These codes are called component codes. Each such a code $\mathcal{C}_s$, $1 \leq s \leq D$, has length η_s and we assign to it a coloring of the array codeword $\mathcal{A}$. Position j of the code $\mathcal{C}_s$ is the binary sum of all positions in $\mathcal{A}$ colored with color j by the s-th coloring. Assume that the size of the burst is B. The first code is a $(B+\delta_1)$-burst-correcting code, $\delta_1 \geq 0$. This code finds and corrects the shape of the error in the codeword of $\mathcal{C}_1$. The error that $\mathcal{C}_1$ corrects can occur in a few positions of the array codeword $\mathcal{A}$. It might also have different shapes in $\mathcal{A}$, but the erroneous positions in $\mathcal{A}$ have the colors of the positions which were in error in the codeword of $\mathcal{C}_1$. The s-th component code, $2 \leq s \leq D$, is a $(B+\delta_s)$-burst-locator code, $\delta_s \geq \delta_1$ (usually, δ_s will be the same integer for all $2 \leq s \leq D$). Burst-locator codes were discussed in [3] and are designed to find the location of a one-dimensional burst which its shape is given up to a cyclic shift. Each of these codes, $\mathcal{C}_s$, provides additional information concerning the positions of the errors, i.e., it reduces the sets of possible locations of errors in $\mathcal{A}$ as were found by $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_{s-1}$. Finally, the last component code finds the actual positions of the burst-error. To execute these tasks the colorings should satisfy a few properties:

- (**p.1**) For the s-th coloring, for each s, $1 \leq s \leq D$, the colors inside a burst of the given shape are distinct integers and the difference between the largest integer and the smallest one is at most $B + \delta_s - 1$.
- (**p.2**) Given the D colorings and a color ν_s, for the s-th coloring, for each $1 \leq s \leq D$, there is at most one position in the array which is colored with the colors $(\nu_1, \nu_2, \ldots, \nu_D)$.
- (**p.3**) Any two positions which are colored with the same color by the first coloring, have colors which differ by a multiple of $B + \delta_s$ by the s-th coloring, for each s, $2 \leq s \leq D$.

The redundancy of the D-dimensional code is the sum of the redundancies of the D component codes. If we use a $(B+\delta_s)$-burst-correcting code for the s-th component code then this code does not need to satisfy (**p.3**). The disadvantage will be that the total redundancy of the multidimensional code will increase. The advantage will be that we will be more flexible in the parameters of the multidimensional code since burst-locator codes are more rare than burst-correcting codes.

Which codes can be used for the coloring method? We start with the code for the first component code. The most efficient codes are those constructed by Abdel-Ghaffar et al. [10] for correction of a b-burst. For a code of length n, the redundancy of the code is $\lceil \log n \rceil + b - 1$. The main disadvantage of these codes is that their existence depends on a sequence of conditions which are not easy to satisfy.

What about the locator codes? We can use locator codes derived from the codes of Abdel-Ghaffar [10], [11] as demonstrated in [3]. The redundancy of a locator code of length n is $\lceil \log n \rceil$, i.e., it does not depend on the length of the burst. But, these locator codes exist only for odd burst length [3]. Component codes with the parameters η_s, b and D, which satisfy (**p.3**) are usually difficult to find. Hence, if we want codes designed especially to fit the parameters η_s, b and D we should compromise on the redundancy of the component codes which will result in larger redundancy of the multidimensional code. The best codes known for this purpose are the Fire codes [12], [13]. A Fire code of length n which corrects a b-burst has redundancy at most $\lceil \log n \rceil + 2b - 1$.

It will be more convenient if each coloring is a linear function of the coordinate indices, i.e., given a position $(i_1, i_2, \ldots, i_D)$, its color for the s-th coloring is defined by

$$\sum_{k=1}^{D} \alpha_k^s i_k$$

where α_k^s is a constant integer which depends on the coloring s and the shape of the D-dimensional cluster. Such a coloring will be called a *linear coloring*. With a linear coloring we associate a *coloring matrix* A_D, where $(A_D)_{s,k} = \alpha_k^s$. It is easy to verify that property (**p.2**), is fulfilled for a linear coloring if and only if the coloring matrix is invertible.

Now we will apply the coloring method on two types of errors, semi-crosses and crosses with arms of length one. If we will try to correct an error of either type by correcting a box-error which inscribes it then the excess redundancy will be exponential in D. The lower bound on the excess redundancy is linear in D and our code will have slightly larger excess redundancy. For simplicity we will assume for the rest of this section that all the edges of our array are equal to n. We use the notation $(\times n)^D$ to denote $\underbrace{n \times \cdots \times n}_{D \text{ times}}$.

A. Semi-crosses with arms of length one

We define the colorings by the coloring matrix, which is a $D \times D$ matrix $A = \{a_{ij}\}_{1 \le i,j \le D}$

$$a_{1k} = k, \quad 1 \le k \le D \ .$$

For each s, $2 \le s \le D$ we define

$$a_{sk} = k, \quad 1 \le k < s \ ,$$
$$a_{sk} = k - D - 1, \quad s \le k \le D \ .$$

The s-th color, $1 \le s \le D$, of position $(i_1, i_2, \ldots, i_D)$ in the array is given by

$$c^s_{(i_1,i_2,\ldots,i_D)} = \sum_{k=1}^{D} a_{sk} i_k.$$

Using these colorings we obtain the following result. We present here its proof in order to demonstrate how the coloring method works. The proofs for all other colorings is similar as they all satisfy properties (**p.1**), (**p.2**), and (**p.3**).

Theorem 3: For any given even D, there exists a code which corrects any D-dimensional error confined to a semi-cross burst with radius one in an $(\times n)^D$ cube and its redundancy is at most $\lceil \log n^D \rceil + 2D \lceil \log (D+1)) \rceil + D$.

Proof: One can verify that the three coloring properties hold. Therefore, given the set of erroneous colors by the first coloring, according to property (**p.3**) the shape of the burst in all other colorings is known up to cyclic permutation. Therefore, for $2 \le s \le D$, the burst-locator code can find the locations of the erroneous colors in the s-th coloring. Then, for each error in the multidimensional array, its set of colors by each coloring is known and according to property **p.2** it is possible to find the error location in the array. ■

Better redundancy is obtained if we slightly change the coloring and define a nonlinear coloring. The s-th color, $1 \le s \le D$, of position $(i_1, i_2, \ldots, i_D)$ is given by

$$c^s_{(i_1,i_2,\ldots,i_D)} = \left(\sum_{k=1}^{D} a_{sk} i_k \right) \bmod (n(D+1)).$$

As a consequence we have the following theorems.

Theorem 4: For any given even D, there exists a code which corrects any D-dimensional error confined to a semi-cross burst with radius one in an $(\times n)^D$ cube and its redundancy is at most $\lceil \log \ n^D \rceil + D \lceil \log (D+1)) \rceil + D$.

If we use the Fire codes [12], [13] as locator codes we obtain the following theorem.

Theorem 5: For any given D and n, there exists a code which corrects any D-dimensional error confined to a semi-cross burst with radius one in an $(\times n)^D$ cube and its redundancy is at most $\lceil \log n^D \rceil + 2D^2 + D \lceil \log \ (D+1) \rceil + D$.

B. Crosses with arms of length one

We define the colorings by the coloring matrix, which is a $D \times D$ matrix $A = \{a_{ij}\}_{1 \le i,j \le D}$

$$a_{ij} = ij \bmod (2i(D-i+1)+1) \ ,$$
$$a_{ij} \in \{-i(D-i+1), \ldots, -1, 0, 1, \ldots, i(D-i+1)\}.$$

The first color of position $(i_1, i_2, \ldots, i_D)$ in the array is given by

$$c^1_{(i_1,i_2,\ldots,i_D)} = \sum_{k=1}^{D} a_{1k} i_k.$$

The s-th color, $2 \le s \le D$, of position $(i_1, i_2, \ldots, i_D)$ in the array is given by

$$c^s_{(i_1,i_2,\ldots,i_D)} = \left(\sum_{k=1}^{D} a_{sk} i_k \right) \bmod (2s(D-s+1)n).$$

Theorem 6: There exists a code which corrects any D-dimensional error confined to a Lee sphere burst with radius one in an $(\times n)^D$ cube and its redundancy is at most $\lceil \log n^D \rceil + 2D \lceil \log D \rceil$.

Theorem 7: For any given D and n, there exists a code which corrects any D-dimensional Lee sphere burst with radius one in an $(\times n)^D$ cube and its redundancy is at most $\lceil \log n^D \rceil + 2D^2 + 2D \lceil \log D \rceil$.

Remark 4: For specific values of D, i.e., when $2D+1$ is a prime number, $11 \le 2D+1 \le 10000$, this construction can be slightly improved.

V. Clusters with Limited Weight

When a certain area suffers from an error event we might expect that not all the positions will be in error. Hence, it seems that practically, we would expect to correct a cluster with a limited weight. In this section we will consider the case where the weight of the cluster is at most two.

A. Semi-crosses

We start by correcting an error with weight at most two in a D-dimensional semi-cross with arms of length one.

Construction C: Let α be a primitive element in GF(2^m) for $2^m - 1 \ge \prod_{\ell=1}^{D} n_\ell$. Let $d = \lceil \log D \rceil$ and $\mathbf{i} = (i_1, i_2, \ldots, i_D)$, where $0 \le i_\ell \le n_\ell - 1$. Let $\mathcal{H}$ be a $(2d) \times D$ parity-check matrix of a double-error correcting BCH code (or its shortened code). We construct the following $n_1 \times n_2 \times \cdots \times n_D \times (m+2d+1)$ parity check matrix H:

$$h_{\mathbf{i}} = \begin{bmatrix} 1 \\ \mathcal{H}\mathbf{i}^T \bmod 2 \\ \alpha^{\sum_{j=1}^{D} i_j (\prod_{\ell=j+1}^{D} n_\ell)} \end{bmatrix},$$

for all $\mathbf{i} = (i_1, i_2, \ldots, i_D)$, where $0 \le i_\ell \le n_\ell - 1$.

Theorem 8:

- The code constructed in Construction C can correct any error of weight at most two inside a semi-cross with arms of length one in an $n_1 \times n_2 \times \cdots \times n_D$ array codeword.
- The code constructed by Construction C has redundancy which is greater by at most two from the trivial lower bound on the redundancy.

Proof: The first part of the Theorem is an immediate consequence from the decoding procedure and the second part is easily verified. The decoding is very similar to the one of Construction A. If the received syndrome is the all zero vector then no error occurred. The first bit of the syndrome indicates whether one or two errors occurred. If the first bit is one, then one error occurred, and its location can be found by the rest of the syndrome. Otherwise, two errors occurred. The two errors can be of the form $\mathbf{i}_1 = (i_1, \ldots, i_D)$ and $\mathbf{i}_2 = (i_1, \ldots, i_{j-1}, i_j + 1, i_{j+1}, i_D)$ or $\mathbf{i}_1 = (i_1, \ldots, i_{k-1}, i_k + 1, i_{k+1}, i_D)$ and $\mathbf{i}_2 = (i_1, \ldots, i_{j-1}, i_j + 1, i_{j+1}, i_D)$. In the first case, the next $2d$ bits of the syndrome are the j-th column of the matrix $\mathcal{H}$, and in the second case the next $2d$ bits are the sum of the j-th and k-th columns of the matrix $\mathcal{H}$. Since $\mathcal{H}$ is a parity check matrix of a double error-correcting code it is possible to distinguish between these cases and know the shape of the error. Thus, as in Construction A, the last m bits of the syndrome indicate the location of the error. ∎

Construction C is generalized for a semi-cross with arms of length R with some extra redundancy in Construction D which follows.

Construction D: Let α be a primitive element in GF(2^m) for $2^m - 1 \geq \prod_{\ell=1}^{D} n_\ell$. Let t be the smallest integer such that $2^t - 1 \geq 2RD$. Let $\mathcal{H}$ be a $(4t) \times (2^t - 1)$ parity-check matrix of a four-error-correcting BCH code. Let $\Pi = \{\hat{H}^1, \hat{H}^2, ..., \hat{H}^D\}$ be a set of disjoint subsets of columns of size $2R$ from $\mathcal{H}$, where $\hat{H}^i = [\hat{h}_0^i, \ldots, \hat{h}_{2R-1}^i]$. We construct the following $n_1 \times n_2 \times \cdots \times n_D \times (m + 4t + 1)$ parity-check matrix:

$$h_{\mathbf{i}} = \begin{bmatrix} 1 \\ \sum_{\ell=1}^{D} \hat{h}^{\ell}_{i_\ell \bmod 2R} \bmod 2 \\ \alpha^{\sum_{j=1}^{D} i_j (\prod_{\ell=j+1}^{D} n_\ell)} \end{bmatrix},$$

for all $\mathbf{i} = (i_1, i_2, \ldots, i_D)$, where $0 \leq i_\ell \leq n_\ell - 1$.

Theorem 9:

- The code constructed in Construction D can correct any error of weight at most two inside a semi-cross with arms of length R in an $n_1 \times n_2 \times \cdots \times n_D$ array codeword.
- The code constructed by Construction D has excess redundancy $4\lceil \log\, D + \log\, R \rceil + 5$ compared to a trivial lower bound $2\log\, D + 2\log\, R - 1$.

Remark 5: Two errors inside a two-dimensional square, of a D-dimensional array, with edge of length R can be viewed as an error with weight two inside a semi-cross with arms of length $R - 1$. Hence, Construction D can be used to correct the related error.

B. Crosses

The idea of correcting an error with weight two in a cross is a modification of the one for a semi-cross. The two directions in which the two errors occurred are revealed exactly as in the semi-cross. The difference is that in the semi-cross the directions are only positive, while in the cross they can be either positive or negative. We will demonstrate how to solve the problem in the case when the cross has arms with length one. Similar solution is given for longer arms. It appears that we only have to find whether the two directions have the same sign or not, reducing the number of cases from four to two.

Construction E: Let α be a primitive element in GF(2^m) for $2^m - 1 \geq \prod_{\ell=1}^{D} n_\ell$. Let t be the smallest integer such that $2^t - 1 \geq 4D$. Let $\mathcal{H}$ be a $(4t) \times (2^t - 1)$ parity-check matrix of a four-error-correcting BCH code. Let $\Pi = \{\hat{H}^1, \hat{H}^2, ..., \hat{H}^D\}$ be a set of disjoint subsets of columns of size 4 from $\mathcal{H}$, where $\hat{H}^i = [\hat{h}_0^i, \hat{h}_1^i, \hat{h}_2^i, \hat{h}_3^i]$. We construct the following $n_1 \times n_2 \times \cdots \times n_D \times (m + 4t + 2)$ parity-check matrix:

$$h_{\mathbf{i}} = \begin{bmatrix} 1 \\ \sum_{\ell=1}^{D} \hat{h}^{\ell}_{i_\ell \bmod 4} \bmod 2 \\ \lfloor \frac{\sum_{j=1}^{D} i_j}{2} \rfloor \bmod 2 \\ \alpha^{\sum_{j=1}^{D} i_j (\prod_{\ell=j+1}^{D} n_\ell)} \end{bmatrix},$$

for all $\mathbf{i} = (i_1, i_2, \ldots, i_D)$, where $0 \leq i_\ell \leq n_\ell - 1$.

Theorem 10: The code constructed in Construction E can correct any error of weight at most two inside a cross with arms of length one in an $n_1 \times n_2 \times \cdots \times n_D$ array codeword.

VI. Conclusion

We have considered various types of errors in a D-dimensional array, where D is a large integer. Given an error pattern we have constructed codes for which the redundancy and the excess redundancy are relatively small. The excess redundancy is much smaller than the one obtained from a construction which produces a code correcting a box-error which contains the given cluster. The most immediate future goal will be to construct D-dimensional codes which correct a cluster error of size b (whose shape is not a D-dimensional box), for large b, with asymptotically optimal excess redundancy.

Acknowledgment

The work of Eitan Yaakobi was supported in part by the University of California Lab Fees Research Program, Award No. 09-LR-06-118620-SIEP. Tuvi Etzion was supported by the United States-Israel Binational Science Foundation (BSF), Jerusalem, Israel, under Grant No. 2006097.

References

[1] K. A. S. Abdel-Ghaffar, "An information- and coding-theoretic study of bursty channels with applications to computer memories", *Ph.D. dissertation, California Inst. Technol. Pasadena, CA*, June 1986.

[2] K. A. S. Abdel-Ghaffar, R. J. McEliece, H. C. A. van Tilborg, "Two-dimensional burst identification codes and their use in burst correction", *IEEE Trans. on Inform. Theory*, vol. IT-34, pp. 494-504, May 1988.

[3] T. Etzion and E. Yaakobi, "Error-Correction of Multidimensional Bursts", *IEEE Trans. on Inform. Theory*, vol. IT-55, pp. 961-976, March 2009.

[4] S. W. Golomb, "A general formulation of error metrics", *IEEE Trans. Inform. Theory*, vol. IT-15, 425–426, 1969.

[5] S. W. Golomb and L. R. Welch, "Perfect codes in the Lee metric and the packing of polyominos", *SIAM Journal Applied Math.*, vol. 18, pp. 302–317, 1970.

[6] S. Stein, "Packings of R^n by certain error spheres", *IEEE Trans. on Inform. Theory*, vol. IT-30, pp. 356-363, March 1984.

[7] W. Hamaker and S. Stein, "Combinatorial packing of R^3 by certain error spheres", *IEEE Trans. on Inform. Theory*, vol. IT-30, pp. 364-368, March 1984.

[8] S. Stein and S. Szabó, *Algebra and Tiling*, The Mathematical Association of America, 1994.

[9] M. Schwartz and T. Etzion, "Two-dimensional cluser-correcting codes", *IEEE Trans. on Inform. Theory*, vol. IT-51, pp. 2121-2132, June 2005.

[10] K. A. S. Abdel-Ghaffar, R. J. McEliece, A. M. Odlyzko, H. C. A. van Tilborg, "On the existence of optimum cyclic burst-correcting codes", *IEEE Trans. on Inform. Theory*, vol. IT-32, pp. 768-775, Nov. 1986.

[11] K. A. S. Abdel-Ghaffar, "On the existence of optimum cyclic burst-correcting codes over GF(q)", *IEEE Trans. on Inform. Theory*, vol. IT-34, pp. 329-332, March 1988.

[12] P. Fire, "A class of multiple error correcting binary codes for nonindependent errors", *Sylvania Reconnaisance Lab., Mountain View, California, Sylvania Rep. RSL-e-2*, 1959.

[13] W. W. Paterson, *Error-Correcting Codes*, Cambridge University Press, 2005.